APPLICATION FOR UNITED STATES LETTERS PATENT

FOR

# PROGRAMMABLE CHIP SELECT

Inventor(s): Peter R. Munguia
Eric C. Wang

Attorney Docket No.: P16384

Prepared by: Jeffrey B. Huter
Patent Attorney

Intel Corporation
5000 W. Chandler Blvd., CH6-404
Chandler, AZ 85226-3699
Phone: (480) 554-4198
Facsimile: (480) 554-7738

"Express Mail" label number: EV 324059825 US

# PROGRAMMABLE CHIP SELECT

## BACKGROUND

[0001] A computing device may comprise firmware routines to initialize components of the computing device. Upon system startup, the computing device may activate a chip select of a non-volatile memory device having the firmware routines. The computing device may then retrieve and execute the firmware routines from the non-volatile memory device.

## BRIEF DESCRIPTION OF THE DRAWINGS

[0002] The invention described herein is illustrated by way of example and not by way of limitation in the accompanying figures. For simplicity and clarity of illustration, elements illustrated in the figures are not necessarily drawn to scale. For example, the dimensions of some elements may be exaggerated relative to other elements for clarity. Further, where considered appropriate, reference labels have been repeated among the figures to indicate corresponding or analogous elements.

[0003] FIG. 1 illustrates an embodiment of a computing device without a chip select decoder.

[0004] FIG. 2 illustrates an embodiment of a computing device with a chip select decoder.

[0005] FIG. 3 illustrates a startup method that the computing devices of FIG. 1 and FIG. 2 may execute.

## DETAILED DESCRIPTION

[0006] The following description describes techniques for accessing non-volatile memory with encoded and unencoded chip select words. In the following description, numerous specific details such as logic implementations, opcodes, means to specify operands, resource partitioning/sharing/duplication implementations, types and interrelationships of system components, and logic partitioning/integration choices are set forth in order to provide a more thorough understanding of the present invention. It will be appreciated, however, by one skilled in the art that the invention may be practiced without such specific details. In other instances, control structures, gate level circuits and full software instruction sequences have not been shown in detail in order not to obscure the invention. Those of ordinary skill in the art, with the included descriptions, will be able to implement appropriate functionality without undue experimentation.

[0007] References in the specification to "one embodiment", "an embodiment", "an example embodiment", etc., indicate that the embodiment described may include a particular feature, structure, or characteristic, but every embodiment may not necessarily include the particular feature, structure, or characteristic. Moreover, such phrases are not necessarily referring to the same embodiment. Further, when a particular feature, structure, or characteristic is described in connection with an embodiment, it is submitted that it is within the knowledge of one skilled in the art to effect such feature, structure, or characteristic in connection with other embodiments whether or not explicitly described.

[0008] Embodiments of the invention may be implemented in hardware, firmware, software, or any combination thereof. Embodiments of the invention may also be implemented as instructions stored on a machine-readable medium, which may be read and executed by one or more processors. A machine-readable medium may include any mechanism for storing or transmitting information in a form readable by a machine (e.g., a computing device). For example, a machine-readable medium may include read only memory (ROM); random access memory (RAM); magnetic disk storage media; optical storage media; flash memory devices; electrical, optical, acoustical or other forms of propagated signals (e.g., carrier waves, infrared signals, digital signals, etc.); and others. Further, firmware, software, routines, instructions may be described herein as performing certain actions. However, it should be appreciated that such descriptions are merely for convenience and that such actions in fact result from computing devices, processors, controllers, or other devices executing the firmware, software, routines, instructions, etc.

[0009] Example embodiments of a computing device 100 are shown in FIG. 1 and FIG. 2. As illustrated, the computing device 100 may comprise one or more processors 102. The processors 102 may perform actions in response to executing instructions of an operating system 104, application 106, basic input/output system (BIOS) firmware 108, and/or some other software or firmware module.

[0010] The computing device 100 may further comprise a chipset 110 that is coupled to the processors 102 via a processor bus. The chipset 110 may

comprise one or more integrated circuit packages or chips that couple the processors 102 to other components of the computing device 100. In particular, the chipset 110 may comprise a volatile memory controller 112 that is coupled to volatile memory 114 via a memory bus. The chipset 110 may further comprise a non-volatile memory controller 116 that is coupled to a non-volatile memory 118 via an address/data bus 120, chip select lines $122_0$, $122_1$ ... $122_N$, and/or other bus and control lines (not shown) such as, for example, row address strobes, column address strobes, write enables, etc.

[0011] Embodiments of the computing device 100 are illustrated in FIG. 1 and FIG. 2 with separate volatile and non-volatile memory controllers 112, 116. However, other embodiments may comprise a single memory controller to access both the volatile memory 114 and non-volatile memory 118. In yet another embodiment, the computing device 100 may be implemented without volatile memory 114 and a volatile memory controller 112.

[0012] The volatile memory 114 may comprise volatile memory devices (not shown) having addressable storage locations that may be read from and/or written to. The volatile memory devices may comprise one or more different volatile memory types such as, for example, RAM (Random Access Memory) devices, SRAM (Static RAM) devices, DRAM (Dynamic RAM Memory) devices, SDRAM (synchronous DRAM) devices, DDR (Double Data Rate) SDRAM devices, etc.. The non-volatile memory 118 may comprise non-volatile memory devices 124 having addressable storage locations that may be read from and/or written to. The non-volatile memory device 124 may comprise one or more

different non-volatile memory types such as, for example, Flash memory devices, ROM (read only memory) devices, PROM (programmable read only memory) devices, EPROM (eraseable PROM) devices, EEPROM (electrically eraseable PROM) devices, Ferroelectric memory devices, battery-backed memory devices, etc.

[0013] The non-volatile memory controller 116 may comprise a programmable configuration store 126. The computing device 100 may update or program the configuration store 126 to define the operation of the non-volatile memory controller 116. In one embodiment, the configuration store 126 may configure the non-volatile memory controller 116 for an encoded chip select mode or an unencoded chip select mode. Moreover, the configuration store 126 may configure the non-volatile memory controller 116 for various non-volatile memory configurations. For example, the non-volatile memory controller 116 via the configuration store 126 may support multiple memory types, memory capacities, memory timings, etc.

[0014] In one embodiment, the configuration store 126 may default the non-volatile memory controller 116 to an encoded chip select mode. The configuration store 126 may further default the non-volatile memory controller to a memory configuration that assumes a non-volatile memory device 118 coupled to a predetermined chip select line 122 (e.g. chip select line $122_0$), that the non-volatile memory device 118 consists of a minimum storage capacity (e.g. 1 megabyte) supported by the non-volatile memory device 124, that maps a boot code nub address to a boot code nub 128 stored in a predetermined non-volatile

memory device 118, and that rolls over addresses to the predetermined non-volatile memory device 124 to ensure all addresses target the predetermined non-volatile memory device 124 until configured otherwise.

[0015] The non-volatile memory controller 116 may comprise an address decoder 130 to decode an address into a chip select word that selects a non-volatile memory device 124 to service a transaction, into a row address that selects the appropriate row of the selected non-volatile memory device 124, and a column address that selects the appropriate column of the selected row. In response to the unencoded chip select mode, the non-volatile memory controller 116 may generate an unencoded chip select word 132 (FIG. 1). The address decoder 130 may generate the unencoded chip select word 132 such that the unencoded chip select word 132 activates only a single chip select line 122 and deactivates the other chip select lines 122. In such an embodiment, the active chip select line 122 may select an associated non-volatile memory device 124 or a group of associated non-volatile memory devices 124 to service a transaction and the inactive chip select lines 122 may deselect the other non-volatile memory devices 124 of the non-volatile memory 118 that are not to service the transaction.

[0016] In response to the encoded chip select mode, the address decoder 130 may further generate an encoded chip select word 134 (FIG. 2) and provide the encoded chip select word 134 to a chip select decoder 136. The chip select decoder 136 may generate, from the received encoded chip select word 134, an unencoded chip select word 132 that activates only a single chip select line 122

and that deactivates the other chip select lines 122. In such an embodiment, the active chip select line 122 may select an associated non-volatile memory device 124 or a group of associated non-volatile memory devices 124 to service a transaction and the inactive chip select lines 122 may deselect the other non-volatile memory devices 124 of the non-volatile memory 118 that are not to service the transaction.

[0017] In an embodiment that supports up to eight non-volatile memory devices 124, the address decoder 130 may generate unencoded and encoded chip select words 132, 134 according to TABLE 1. Further, the chip select decoder 136 may generate the unencoded chip select words 134 in response to the receiving corresponding encoded chip select words 134 from the address decoder 130. In particular, the address decoder 130 and chip select decoder 136 may generate CS_Word_X to select the non-volatile memory device 124 coupled to the chip select line $122_X$. Moreover, a binary 0 in TABLE 1 indicates an inactive bit of the chip select word and a binary 1 indicates an active bit of the chip select word. An active bit may correspond to a high signal and an inactive bit may correspond to a low signal. However, other signal encodings may be used such as, for example, representing an active bit with a low signal and an inactive bit with a high bit.

[0018]

| | Encoded Chip Sel cts CS[2:0] | Unencoded Chip Selects CS[7:0] |
|---|---|---|
| CS_Word_0 | 001b | 0000_0001b |
| CS_Word_2 | 010b | 0000_0010b |
| CS_Word_2 | 011b | 0000_0100b |
| CS_Word_3 | 100b | 0000_1000b |
| CS_Word_4 | 101b | 0001_0000b |
| CS_Word_5 | 110b | 0010_0000b |
| CS_Word_6 | 111b | 0100_0000b |
| CS_Word_7 | 000b | 1000_0000b |

TABLE 1

[0019] As illustrated in TABLE 1, the three lowest order bits [2:0] of the encoded chip select word CS_Word_0 and the three lowest order bits [2:0] of the unencoded chip select word CS_Word_0 are the same and include a single active bit. Assuming a valid platform, the address decoder 130 may select the non-volatile memory device 124 coupled to chip select line $122_0$ by generating the unencoded chip select word CS_Word_0 or the encoded chip select word CS_Word_0. For example, if the computing device 100 does not include a chip select decoder 136 (See, FIG. 1), then the single active bit of the unencoded or encoded chip select word CS_Word_0 may activate the chip select line $122_0$ and may select the corresponding non-volatile memory device 124. Similarly, if the computing device 100 includes a chip select decoder 136 (See, FIG. 2), then the chip select decoder 136 may generate the unencoded chip select word

CS_Word_0 from a portion (e.g. the three lowest order bits CS[2:0]) of the unencoded chip select word CS_Word_0 or the encoded chip select word CS_Word_0 generated by the address decoder 130. The single active bit of the unencoded chip select word CS_Word_0 generated by the chip select decoder 136 may then activate the chip select line $122_0$ and may select the corresponding non-volatile memory device 124.

[0020] In one embodiment, the BIOS firmware 108 may comprise routines which the computing device 100 may execute during system startup in order to initialize the processors 102, chipset 110, and other components of the computing device 100. Moreover, the BIOS firmware 108 may comprise routines or drivers which the computing device 100 may execute to communicate with one or more components of the computing device 100. In one embodiment, the BIOS firmware 108 may comprise routines to configure the non-volatile memory controller 116 and may comprise a boot code nub 128 to configure components of the computing device 100. In one embodiment, the boot code nub 128 may update the configuration store 126 to program whether the address decoder 130 generates encoded or unencoded chip select words 132, 134. The BIOS firmware 108 may further update the configuration store 126 to indicate which chip select lines $122_0$, $122_1$ ... $122_N$ are coupled to non-volatile memory devices 118 and to indicate the capacity of each non-volatile memory device 118.

[0021] As depicted in FIG. 1, the boot code nub 128 may reside in a non-volatile memory device 124 associated with a predetermined chip select line (e.g. chip select line $122_0$). Moreover, the computing device 100 of FIG. 1 depicts that

the BIOS firmware 108 may span more than one non-volatile memory device 124. In one such embodiment, the computing device 100 may executed the boot code nub 128 in order to configure the non-volatile memory controller 118 and enable access to the other non-volatile memory devices 124 and startup routines of the BIOS firmware 108 that reside in the other non-volatile memory devices 124. However, in other embodiments, the BIOS firmware 108 may reside in a single non-volatile memory device 124.

[0022] The chipset 110 may further comprise input/output (I/O) controllers (not shown) coupled to I/O devices 138 (e.g. a mouse, keyboard, video controller, network interface controller, hard disk, floppy disk, wireless receivers, wireless transmitters, etc.) via buses. For example, the chipset 110 may comprise peripheral component interconnect (PCI) controllers, accelerated graphics port (AGP) controllers, universal serial bus (USB) controllers, low pin count (LPC) bus controllers, and/or other input and/or output (I/O) controllers to control and transfer data via the respective buses.

[0023] Referring now to FIG.3, there is shown a startup method that the computing device 100 of FIG. 1 or FIG. 2 may perform in response to a system power-up, a system reset, or some other event. The processor 102 in block 200 may jump to a boot code nub address (e.g. FFFF_FFFFh). The address decoder 130 may decode the boot code nub address and generate a chip select word 132, 134 (e.g. unencoded or encoded chip select word CS_Word_0) having a single active bit to select the predetermined non-volatile memory device 124 that comprises the boot code nub 128 in block 202. The address decoder 130 may

further generate a row address and column address which the non-volatile memory controller 116 may use to retrieve the portion of the boot code nub 128 stored at the boot code nub address.

[0024] If the computing device 100 does not have a chip select decoder 136, then the active bit of the encoded or unencoded chip select word 132, 134 generated by address decoder 130 may select the predetermined non-volatile memory device 124 that comprises the boot code nub 128 in block 204. However, if the computing device 100 comprises the chip select decoder 136, then the chip select decoder 136 in block 206 may decode the encoded or unencoded chip select word 132, 134 of the address decoder 130 and may generate an unencoded chip select word (e.g. unencoded chip select word CS_Word_0) having a single active bit to select the predetermined non-volatile memory device 124 that comprises the boot code nub 128. The active bit of the unencoded chip select word generated by chip select decoder 136 may select in block 208 the predetermined non-volatile memory device 124 that comprises the boot code nub 128.

[0025] The processor 102 in block 210 may retrieve and execute the boot code nub 128 from the selected non-volatile memory device 124. In one embodiment, the processor 102 may configure components of the computing device 100 such as, for example, the non-volatile memory controller 124 in response to executing the boot code nub 128. In particular, the processor 102 may update the configuration store 126 for an encoded chip select mode if the processor 102 determines that the computing device 100 comprises the chip

select decoder 136. Alternatively, the processor 102 may update the configuration store 126 for an unencoded chip select mode if the processor 102 determines that the computing device 100 does not include a chip select decoder.

[0026] Moreover, processor 102 may update the configuration store 126 to configure the non-volatile memory controller 116 for the types and capacities of the non-volatile memory devices 124 detected by the processor 102 in response to executing the boot code nub 128. The processor 102 may further update the configuration store 126 to disable rolling addresses over to the predetermined non-volatile memory device 124 that comprises the boot code nub 128. After configuring the non-volatile memory controller 116, the processor 102 in response to executing the boot code nub 128 or the BIOS firmware 108 may reassign unused chip select pins for other purposes, such as, a General Purpose Event (GPE).

[0027] While certain features of the invention have been described with reference to example embodiments, the description is not intended to be construed in a limiting sense. Various modifications of the example embodiments, as well as other embodiments of the invention, which are apparent to persons skilled in the art to which the invention pertains are deemed to lie within the spirit and scope of the invention.